CCN 中基于业务分类和节点分区的混合缓存机制 *

刘贵财，兰巨龙

(国家数字交换系统工程技术研究中心，郑州 450002)

摘　要：针对内容中心网络(content centric networking，CCN)中不同业务内容的合理放置问题，提出了一种基于业务分类和节点分区的混合缓存机制。根据不同的业务特征，设计了差异化的缓存策略。对于流媒体点播业务，采用基于流行度的推拉式缓存，实现其在边缘网络的按序存储；对于非流媒体共享内容，采用基于 hash 的显式缓存，实现其在核心网络的单一副本放置。仿真结果表明，与经典算法相比，该机制提高了缓存命中率和跳数减少率，降低了平均请求时延。

关键词：内容中心网络；网内缓存；节点角色；业务类型

中图分类号：TP393　　doi: 10.3969/j.issn.1001-3695.2017.10.1012

Hybrid CCN caching scheme based on service classification and node partition

Liu Guicai, Lan Julong

(*National Digital Switching System Engineering Technological Research Center*, *Zhengzhou* 450002, *China*)

Abstract: Aiming at the reasonable placement of different business contents in content centric networking (CCN) , this paper proposed a hybrid caching scheme based on service classification and node partition (SCNP) . For streaming media-on-demand services, this scheme used push-pull cache strategy based on popularity to achieve its sequential storage in the edge network. For non-streaming contents, this scheme used hash-based explicit cache strategy to implement its placement of single copy in the core network. Simulation results show that compared with the classic cache algorithms, this scheme can improve the cache hit ratio and the hop reduction ratio, and reduce the average request delay.

Key Words: content centric networking (CCN); in-network caching; node role; traffic type

0　引言

随着流媒体及用户自产生等业务的快速发展，网络应用模式逐渐转变为海量内容的分发与获取。而当前互联网基于主机的端到端通信模式缺乏对内容分发的原生支持，这种传统网络通信模式与当前应用需求的不匹配在带宽开销、内容获取时延及用户体验质量等方面严重影响了网络性能。为此，Jacobson 等人革命式地设计了一种新型网络体系结构——内容中心网络(content centric networking, CCN)[1~3]。通过在网络中内置缓存并采用基于内容名的路由，使网络通信模式从以主机为中心转变为以内容为中心，显著提升了网络的内容分发性能。其中，缓存策略的合理设计是提升该性能的关键。

CCN 默认采用 LCE(leave copy everywhere)[4]策略。该策略在沿途所有节点缓存应答内容副本，使得缓存冗余严重，缓存替换频繁，严重降低了缓存网络的整体性能。为此，相关研究者提出了一些改进策略。文献[5]提出了 LCD 策略，数据应答时，在命中节点的下一跳缓存内容副本，随着同一内容请求的不断到达，缓存内容逐渐被复制到用户侧节点。文献[6]提出了 ProbCache 策略，沿途节点以概率缓存内容，且节点距离用户越近，缓存概率越大。文献[7]提出了 Betw 策略，数据包沿请求路径返回时，只在沿途最重要节点缓存；该策略减少了内容请求跳数，但内容在高中心性节点上集中缓存导致该类节点缓存替换频繁，不利于缓存性能的进一步提升。文献[8]提出了一种基于节点社团重要度的缓存策略，将内容缓存在沿途社团内最重要节点，实现缓存内容在空间上的合理分布。文献[9]提出了 MAGIC 策略，根据内容流行度和节点到达源服务器的跳数计算缓存收益，数据应答时，将内容缓存在沿途缓存收益最大的节点上。上述策略大多从内容流行度和节点重要度的角度对缓存策略进行优化设计，而缺乏对业务类型及节点角色区分的考虑。文献[10]根据可靠性和实时性，对内容进行业务类型划分，并设计了差异化的内容请求模式；但没有给出缓存策略的设计方案。文献[11]提出一种基于业务类型的多样化内容分发机制，依据业务请求特征的不同，分别设计了不同的数据分发模式和沿途缓存策略，实现了内容传输与业务类型的匹配；但该机制

基金项目：国家“863”计划资助项目（2015AA016102）

作者简介：刘贵财（1992-），男，河南桐柏人，硕士研究生，主要研究方向为宽带信息网络（15333825619@163.com）；兰巨龙（1962-），男，总工程师，教授，博导，博士，主要研究方向为新一代信息网络关键理论与技术.

设计的缓存策略差别有限，不能很好地满足不同业务类型的差异化缓存需求。文献[12]提出一种基于节点角色划分的混合缓存机制，通过将网络划分成不同的缓存域，并在不同的缓存域内执行差异化的缓存策略，实现用户体验质量与网络资源利用率之间的有效平衡；但是该机制在缓存策略的设计上缺乏对于业务类型的考虑。

笔者认为，对于不同的业务类型及节点角色，设计适合该类业务特性且充分利用节点空间资源的缓存策略，能够使缓存网络结构更合理、性能更高效。为此，本文提出了一种基于业务分类和节点分区的混合缓存机制(hybrid caching scheme based on service classification and node partition, SCNP)。该机制将缓存网络划分为边缘网络和核心网络，并针对流媒体点播业务与非流媒体共享内容的不同特征设计差异化的缓存策略。在边缘网络内，采用基于流行度的推拉式缓存，实现流媒体点播业务内容的按序存储；在核心网络内，采用基于 hash 的显式缓存，实现非流媒体共享内容的合理放置与快速查找。

1 系统模型

本文将缓存系统建模为一个单个 ISP 网络，如图 1 所示。根据 ISP 网络中的自然分域将其划分成一个核心网络和多个边缘网络，其中骨干域划分为核心网络，每个非骨干域划分为一个边缘网络。本缓存系统中共有五种路由器，即下边界路由器、边缘路由器、上边界路由器、核心路由器和网关。其中，下边界路由器、边缘路由器和上边界路由器位于边缘网络，属于边缘网络节点；核心路由器和网关位于核心网络，属于核心网络节点。用户通过下边界路由器接入边缘网络，边缘网络通过上边界路由器与核心网络相连。本文假设所有的内容存储在一个大型服务器内，网关通过一条超宽带链路与该服务器相连。另外，本文默认采用单路径路由进行兴趣包转发，且不考虑节点和链路失效问题。

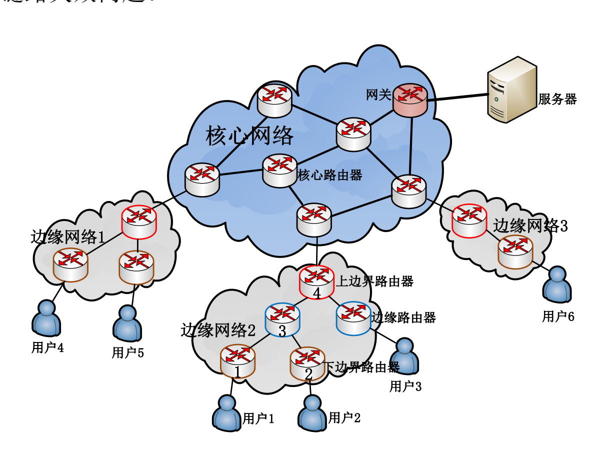


图 1 网络拓扑图

2 SCNP 缓存机制

本文将 CCN 网络中需要被缓存的业务分为两类：a）流媒体点播业务，如视频点播业务，该类业务内容文件大，后续共享程度高，其传输需要消耗大量的带宽资源，且对于时延的要求高；b）非流媒体共享内容，如文字新闻、静态图片等，该类业务内容文件小、数量大，后续共享程度也较高，但对于带宽和时延没有明显要求。

针对流媒体点播业务与非流媒体共享内容的不同特性，本文分别设计了与其业务特征相适应的缓存策略。对于流媒体点播业务，优先选择在边缘网络缓存，采用基于流行度的推拉式缓存策略，实现流媒体点播业务内容在边缘网络内的按序储存，降低该类业务内容的大文件重复传输对于网络带宽资源的吞噬性消耗及用户获取该类业务内容的时延，提高用户的视频和音频体验质量。对于非流媒体共享内容，只在核心网络缓存，采用基于 hash 的显式缓存策略，实现非流媒体共享内容在核心网络内的单一副本放置与快速查找，同时为流媒体点播业务内容在边缘网络缓存节省更多储存空间。

2.1 基于流行度的推拉式缓存策略

2.1.1 流行度等级

定义 1 内容流行度 P。P 定义为任一节点上，确定时长内某一内容在该节点上的请求次数期望。文献[13]采用大小为 W_{KT} 的滑动窗口对内容 c 在节点 v 上的请求次数进行统计，并根据 EWMA（exponential weighted moving average）模型提出一种内容流行度的计算方法。根据该方法，内容 c 在节点 v 上的流行度 P_v^c 可表示为

$$P_v^c=\frac{p_i^c+(1-\varphi)p_{i-1}^c+(1-\varphi)^2p_{i-2}^c+\cdots+(1-\varphi)^K p_{i-K}^c}{1+(1-\varphi)+(1-\varphi)^2+\cdots+(1-\varphi)^K} \tag{1}$$

其中：p_i^c 为内容 c 在第 i 个窗口内的请求次数；K 为滑动窗口大小；T 为确定时长；$\varphi=\frac{2}{K+1}$ [14]为衰减因子。

本文在下边界路由器对流媒体点播业务内容的请求次数进行统计，实现对该类业务内容流行度的感知。为了区分流行度级别，本文设置了一个流行阈值(popularity threshold, PT)，流行度超过该阈值的流媒体点播业务内容，被视为流行度等级高。为了实现缓存节点对业务类型及流媒体点播业务流行度等级的识别与判断，本文在兴趣包与数据包中分别添加了一个 2 bit 的判别(judgment and identification, JaI)字段。其中，第一个比特为业务类型(type of service, ToS)位，ToS=1 表示流媒体点播业务，ToS=0 表示非流媒体共享内容；第二个比特为流行度等级(level of popularity, LoP)位，对于流媒体点播业务，LoP=1 表示内容流行度等级高，LoP=0 表示内容流行度等级低，对于非流媒体共享内容，该位的值始终设置为 0。

2.1.2 推拉式缓存决策

流媒体点播业务对带宽和时延的要求均较高，因此，该类业务内容对缓存在用户侧节点的需求比较强烈。本节在缓存决策时，仅选择流行度等级高的流媒体点播业务内容缓存在边缘网络；然后在边缘网络内，根据已缓存内容的后续流行度高低，将其推拉式缓存至距离用户远近程度不同的位置，实现节点空间资源利用及缓存效用（网络带宽节省与用户体验质量提高）

的最大化。

内容请求时，用户根据所请求的业务类型设置兴趣包 ToS 位的值，兴趣包到达下边界路由器后，通过判别业务类型和流行度等级，设置 LoP 位的值，从而确定 JaI 字段的值；数据应答时，内容源将兴趣包 JaI 字段的值赋值于数据包 JaI 字段，然后沿相反方向返回内容。对于流媒体点播业务，应答内容从源服务器返回至上边界路由器时，该路由器查看数据包 JaI 字段的 LoP 位，若其值为 1，说明该内容的流行度等级高，则缓存该内容。即仅流行度等级高的流媒体点播业务内容能够缓存在边缘网络，且仅从上边界路由器缓存至边缘网络。

对于边缘网络中已缓存的内容，每一次缓存命中，相应命中内容被拉至下游（用户侧）一跳节点缓存。若该节点的缓存空间已满，则替换掉的内容被推向上游（核心网络侧）一跳节点缓存；若内容从上边界路由器中被替换掉，则转存至核心网络节点，避免一些高流行度流媒体点播业务内容被替换所造成的缓存效用损失。即流媒体点播业务内容缓存至上边界路由器后，随着后续流行度的高低，被逐渐复制到下边界路由器或替换至核心网络。由上述推拉式缓存的特性可知，流行阈值 PT 的设置应适中。若 PT 值太大，一些高流行度流媒体点播业务内容得不到缓存，导致边缘网络中的缓存内容上推至核心网络的速度太慢，从而使已缓存在边缘网络但已不再流行的流媒体点播业务内容长时间占据边缘网络资源，造成边缘网络节点储存空间的极大浪费；若 PT 值太小，则导致上边界路由器缓存替换过于频繁，从而使一些高流行度流媒体点播业务内容还没来得及下拉至下游节点，就已经被替换掉，造成缓存后续利用率低。

图 2 为推拉式缓存的具体案列。假设 C1、C2、C3 为 3 个大小相等的流媒体点播业务内容分块，且 C1、C2、C3 的流行度均超过流行阈值，即对应数据包中 LoP 位的值均为 1，路由器的缓存空间仅能容纳一个内容分块。如图 2(a)所示，用户首先发送对 C1 的请求。由于对应数据包 LoP 位的值为 1，C1 从服务器返回至用户的过程中将在上边界路由器缓存。如图 2(b)所示，用户然后相继发送对 C1、C2 的请求。C1 在上边界路由器命中，被转存至下边界路由器；C2 同图 2(a)中情况，被缓存至上边界路由器。如图 2(c)所示，用户最后相继发送对 C2、C3 的请求。同图 2(b)中 C1，C2 被转存至下边界路由器；C1 从下边界路由器替换掉后，被推至上边界路由器缓存；但随着 C3 请求的到达，C1 将被替换至核心网络节点缓存。

2.2 基于 hash 的显式缓存策略

非流媒体共享内容对带宽和时延的要求不高，因此，相对于流媒体点播业务内容，该类业务内容对缓存在靠近用户位置的需求并不强烈。另外，考虑到边缘网络节点空间资源有限，本文将非流媒体共享内容缓存在相对远离用户的核心网络，为流媒体点播业务内容在边缘网络缓存节省更多的存储空间，实现缓存网络空间资源的合理分配。对于非流媒体共享内容的请求，相应兴趣包到达上边界路由器后，执行 hash 运算，确定目标缓存节点。然后根据 FIB 表转发兴趣包至该节点，若该节点中已缓存有请求内容，则直接返回数据;否则，数据包从源服务器返回时缓存应答内容至该节点。所有的上边界路由器均运行相同的 hash 函数，使来自不同边缘网络的同一内容请求，在核心网络中的缓存节点相同，实现核心网络存储资源的充分利用。Hash 函数设置如下：

$$\text{Hash}(\text{content identification}) \rightarrow v_j, v_j \in V_c \tag{2}$$

其中:输入为请求内容的标志；输出为目标缓存节点 v_j；V_c 为核心网络中的节点集合。从上边界路由器中替换掉的流媒体点播业务内容，也以上述 hash 运算的方式确定其在核心网络中的转存节点。

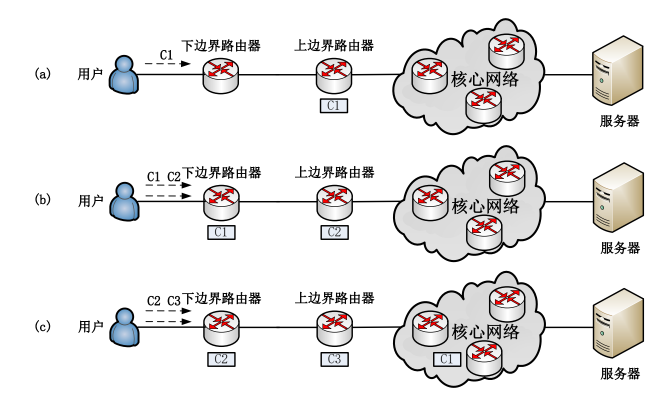


图 2　推拉式缓存示例图

2.3 去冗余功能

如图 1 所示，假设高流行度流媒体点播业务内容分块 C 在某一时刻缓存在上边界路由器 4，且在此后的一段时间内，随着用户 1 和 2 对内容分块 C 请求的不断到达，内容分块 C 最终被下拉至下边界路由器 1 和 2 缓存。假设一段时间后，内容分块 C 不再流行，而随着后续其他流行内容分块的不断下拉缓存，原本缓存在下边界路由器 1 和 2 中的内容分块 C 将被上推至上游路由器。假设内容分块 C 相继从下边界路由器 1 和 2 中被替换掉，则根据上推原则，边缘路由器 3 中将同时缓存内容分块 C 的两个副本，这样将严重浪费边缘网络的缓存空间及降低缓存内容的多样性。因此，为了降低边缘网络中的缓存冗余，在内容分块上推过程中，首先查看路由器中是否已缓存有该内容分块，若有则不再执行缓存操作。内容分块从上边界路由器替换至核心网络节点时，也存在重复缓存的情况，因此，流媒体点播业务内容分块在核心网络缓存时，也执行上述去冗余操作，实现核心网络缓存空间的充分利用。

2.4 算法描述

内容请求时，根据业务类型和流行度等级确定 JaI 字段的值；数据应答时，根据 JaI 值选择内容缓存位置（区域）与缓存策略。若 JaI=00，则在核心网络执行基于 hash 的显式缓存策略；若 JaI=10，则中间节点不执行缓存操作；若 JaI=11，则在边缘网络执行推拉式缓存策略。在将内容分块推至上一跳节点或核心网络节点缓存时，若节点内已缓存有该内容分块，则不再重复储存，降低网络中的缓存冗余。具体过程如算法 1、2 所示。

算法 1　兴趣包(interest packet)处理过程。interest packet 上行转发过程

初始化：content identification (C)、popularity threshold (PT)

(1)用户发送对内容分块 C 的 interest packet 至下边界路由器；

(2)if　下边界路由器缓存有该内容分块　then

(3)　　直接返回数据；

(4) else 查看 interest packet ToS 位的值；

(5)　　if　ToS=1　then

(6)　　　提取内容分块 C 的流行度 P_C，并比较其与流行阀值 PT 的大小；

(7)　　　if　$P_C \le PT$　then

(8)　　　　令 LoP=0，然后转发 interest packet 至内容源；

(9)　　　else 令 LoP=1；然后转发 interest packet 至内容源；

(10)　　else 令 LoP=0，然后转发 interest packet 至上边界路由器进行 Hash 运算，确定目标缓存节点 v_j；

算法 2　数据包(data packet)处理过程。data packet 下行应答过程

(1) 内容源将 interest packet 的 JaI 值赋值于 data packet 的 JaI 字段，并沿相反方向返回 data packet；

(2) if　JaI=11　then

(3)　　if　内容源为核心网络节点或源服务器　then

(4)　　　在沿途上边界路由器缓存内容分块 C；

(5)　　　if　上边界路由器缓存空间已满　then

(6)　　　　进行缓存替换，并对替换掉的内容分块进行 Hash 运算，确定转存节点 v_k；

(7)　　　　发送替换掉的内容分块至节点 v_k；

(8)　　　　if　节点 v_k 已缓存有该内容分块　then

(9)　　　　　丢弃，避免重复缓存；

(10)　　　　else 缓存该内容分块；

(11)　　else if 内容源为边缘网络节点　then

(12)　　　转存内容分块 C 至下游一跳节点；

(13)　　　if　转存节点的缓存空间已满　then

(14)　　　　进行缓存替换，并上推替换掉的内容分块至内容源；

(15)　　　　if　内容源已缓存有该内容分块　then

(16)　　　　　丢弃，避免重复缓存；

(17)　　　　else 缓存该内容分块；

(18) else if　JaI=10　then

(19)　　if 内容源为核心网络节点或源服务器　then

(20)　　　沿途节点不进行缓存操作；

(21)　　else if 内容源为边缘网络节点　then

(22)　　　执行步骤(12)~(17)；

(23) else if　JaI=00　then

(24)　　if 内容源为核心网络节点　then

(25)　　　沿途节点不进行缓存操作；

(26)　　else if 内容源为源服务器　then

(27)　　　在沿途目标缓存节点 v_j 缓存内容分块 C。

3　仿真与性能分析

3.1　仿真环境与参数设置

采用 ndnSIM[15]进行仿真与性能分析。该工具是基于 NS3 的一个 NDN 仿真模块，已实现对 NDN 基本数据单元结构和路由转发流程的模拟。实验拓扑采用 30 个节点的任意网络拓扑结构。采用 K 核分解算法[16]对该网络拓扑进行分区，其中，核心网络节点有 15 个，边缘网络节点中下边界路由器有 9 个。设置 1 个内容服务器，并随机选取一个核心网络节点作为网关与其相连。流媒体点播业务内容为 2 000 个，每个内容划分成大小为 10 KB 的 10 个数据块。非流媒体共享内容为 10 000 个，大小均为 10 KB。内容请求概率服从 Zipf 分布，第 i 个内容的请求概率为 $p(i)=\theta/i^a$，$\theta=\left(\sum 1/i^a\right)^{-1}$。对于流媒体点播业务内容，同一内容的数据块请求按顺序到达，以概率 p=0.9 依次发送对下一数据块的请求。请求到达服从 λ=100 个/s 的泊松分布，其中，流媒体点播业务内容分块和非流媒体共享内容的到达速率分别为 λ_s=70 个/s 和 λ_n=30 个/s。各缓存节点 CS 大小相同，单位为 1 个数据块单元。缓存节点间链路带宽为 100 Mbps，网关与内容服务器间链路带宽为 10 Gbps。仿真时间设为 500 s，滑动窗口大小 K=4，统计间隙 T=5s，流行阈值 PT 取值为第 2 000 个流媒体点播业务内容分块在一个确定时长内的请求次数期望，即 $PT=\lambda_s \cdot T \cdot p(i)$，$i$=2000。节点初始缓存状态为空，缓存替换策略默认为 LRU。其他参数的设置如表 1 所示。

表 1　仿真参数

参数	默认值	变化范围
节点 CS 容量	800	[200,1400]
Zipf 参数 a	1	[0.7,1.3]

3.2　性能指标

将 SCNP 机制与 LCE[4]、DCDS[11]和 Hybrid-HASH-with-LCD[12]进行对比分析.性能评价指标包括缓存命中率(cache hit ratio, CHR)、跳数减少率(hop reduction ratio，HRR)和平均请求时延(average request delay, ARD)。

1）缓存命中率

缓存命中率 CHR 定义为中间节点响应的请求消息占所有请求消息的比例。计算公式如下：

$$CHR=\frac{\omega}{|R|} \tag{3}$$

其中：ω 为中间节点响应的请求消息数量；$|R|$ 为用户发送的请求消息总数。该指标反映了缓存内容的利用效率。

2）跳数减少率

跳数减少率 HRR 定义为相比所有兴趣包都从源服务器端获取响应，部分兴趣包由中间节点响应后，整体传输跳数减少的程度。计算公式如下：

$$HRR=1-\frac{\sum_{r\in R} h_r}{\sum_{r\in R} H_r} \tag{4}$$

其中：h_r 和 H_r 分别为用户实际获取请求内容 C_r 和从源服务器获取 C_r 所需要的路由跳数;R 表示用户发送的请求消息集合。该

指标反映了 SCNP 机制节省带宽的能力。

3）平均请求时延

平均请求时延 ARD 定义为用户获取所有请求响应的时延之和与请求消息总数的比值。计算公式如下：

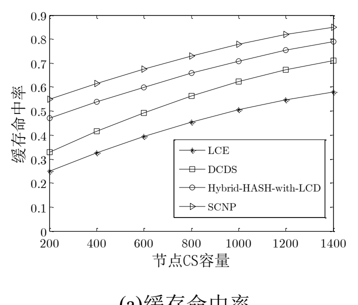
$$ARD=\frac{\sum_{r\in R}T_r}{|R|} \tag{5}$$

其中：T_r 为用户获取请求内容 C_r 的时延。该指标反映了用户请求的响应速度。

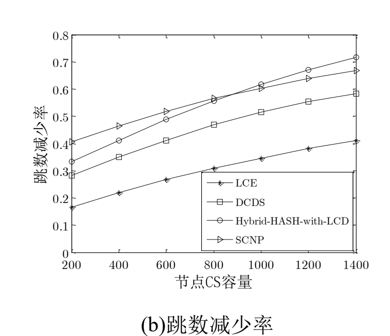
3.3 仿真结果分析

3.3.1 节点 CS 容量的影响

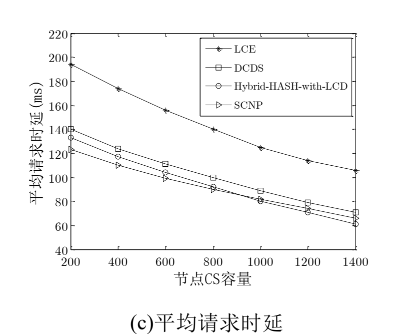
图 3 显示了节点 CS 容量对本实验中四种缓存策略的影响。从图中可以看出，随着节点 CS 容量的增大，四种缓存策略的缓存命中率和跳数减少率增大，平均请求时延减小。节点 CS 容量越大，节点缓存的内容越多，用户请求在中间节点被响应的概率越大（即缓存命中率增大），而用户请求获得响应所需要的跳数越少（即跳数减少率增大），进而平均请求时延减小。如图 3（a）所示，在缓存命中率方面，SCNP 策略的性能始终为最优，相对于次优的 Hybrid-HASH-with-LCD 策略，最大可提升 17.0%。这是由于 SCNP 策略对流媒体点播业务内容和非流媒体共享内容进行分区域缓存，实现了缓存网络空间资源的合理分配与利用，提高了缓存网络的整体命中率。如图 3（b）和（c）所示，在跳数减少率和平均请求时延两个方面，当节点 CS 容量小于 1 000 时，SCNP 策略的性能均为最优，与次优的 Hybrid-HASH-with-LCD 策略相比，分别最大可改善 21.9%和 7.5%，这是因为 SCNP 策略令流媒体点播业务内容独占边缘网络资源，降低了内容请求时延和网络带宽争用；而当节点 CS 容量大于 1 000 时，SCNP 策略的性能低于 Hybrid-HASH-with-LCD 策略，这是因为 SCNP 策略令非流媒体共享内容仅能够在核心网络缓存，当节点 CS 容量继续增大时，边缘网络充足的缓存空间不能被高流行度非流媒体共享内容利用，从而限制了跳数减少率和平均请求时延两个性能指标的进一步改善。而在实际运用中，由于线速处理的需求，节点 CS 容量不应过大。



(a)缓存命中率



(b)跳数减少率

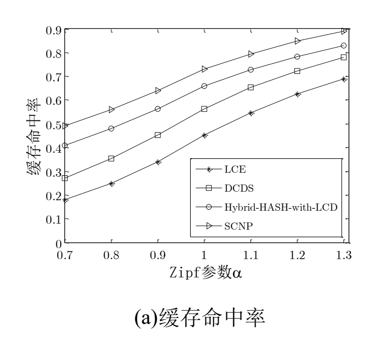


(c)平均请求时延

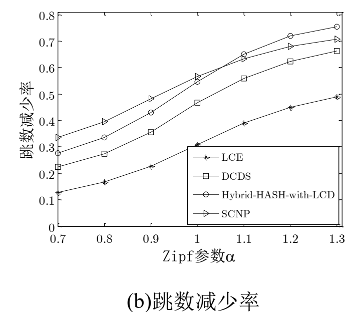
图 3　节点 CS 容量的影响

3.3.2 Zipf 参数 a 的影响

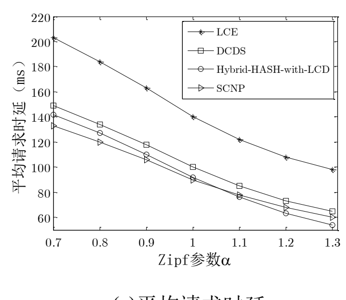
图 4 显示了 Zipf 参数 a 对本实验中四种缓存策略的影响。从图中可以看出，随着参数 a 的增大，四种缓存策略的缓存命中率和跳数减少率均不断增大，而平均请求时延则不断减少。参数 a 越大，内容流行度的分化越明显，缓存少量的热点内容即可满足多数的用户请求，缓存命中的概率增大，进而用户请求获得响应所需的跳数和时延减小。如图 4（a）所示，在缓存命中率方面，SCNP 策略的性能始终优于其他三种策略，与次优的 Hybrid-HASH-with-LCD 策略相比，最大可提升 16.3%。如图 4（b）和（c）所示，在跳数减少率和平均请求时延方面，当参数 a 小于 1.1 时，SCNP 策略的性能优于其他三种策略，与次优的 Hybrid-HASH-with-LCD 策略相比，分别最大可提升 22.2%和 6.3%；当参数 a 大于 1.1 时，SCNP 策略的性能低于 Hybrid-HASH-with-LCD 策略，这是因为随着参数 a 增大，热点内容的数量减少，不同业务内容对边缘网络资源的争用降低，此时，SCNP 策略的分区域缓存方式阻碍了高流行度非流媒体共享内容向用户侧节点的推送，从而限制了用户请求跳数和时延的进一步降低，但当参数 a 较小时，SCNP 策略在缓存命中率、跳数减少率和平均请求时延性能指标上均为最优。



(a)缓存命中率



(b)跳数减少率



(c)平均请求时延

图 4　Zipf 参数 a 的影响

3.4 处理开销

1）通信开销　SCNP 机制在兴趣包和数据包中分别增加了 1 个字段，实现对业务类型和内容流行度等级的判别。但该字段仅为 2 bit 大小，增加的通信开销，可以忽略不计。

2）流行度统计开销　与 LCE、DCDS 和 Hybrid-HASH-with-LCD 相比，SCNP 还增加了内容流行度统计开销。本文仅在下边界路由器且仅对流媒体点播业务内容进行流行度监测，通过限制监测点位置和监测内容类别，降低流行度统计开销。另外，Zhang 等人[17]已经设计出了一种处理速度高达 40 Gbps 的流行度统计算法，并对该算法的复杂度进行了分析。由于下边界路由器的接入流量有限，所以该算法能够很好地满足本文的流行度统计需求。

4　结束语

本文提出一种基于业务分类和节点分区的混合缓存机制 SCNP。针对不同的业务类型，进行分区域缓存。将对带宽和时延要求较高的流媒体点播业务内容，以推拉式缓存的方式放置在边缘网络；而将对带宽和时延没有明显要求的非流媒体共享内容，以 hash 缓存的方式放置在核心网络，实现缓存网络空间资源的合理分配，从而提升缓存网络的性能。仿真测试结果表明，与现有方案相比，SCNP 在缓存命中率、跳数减少率和平均请求时延性能指标上更具优势。后续研究工作将在更加真实的网络环境下，进一步分析 SCNP 机制的性能。

参考文献：

[1] Fang C, Yu F R, Huang T, et al. A survey of green information-centric networking: research issues and challenges [J]. IEEE Communications Surveys & Tutorials, 2015, 17 (3): 1455-1472.

[2] Amadeo M, Campolo C, Quevedo J, et al. Information-centric networking for the internet of things: challenges and opportunities [J]. IEEE Network, 2016, 30 (2): 92-100.

[3] Guan J F, He Y H, Wei Q, et al. A classification-based wisdom caching scheme for content centric networking [C]// Proc of IEEE INFOCOM 2016. 2016: 822-827.

[4] Laoutaris N, Syntila S, and Stavrakakis I. Meta algorithms for hierarchical Web caches [C]// Proc of IEEE International Performance Computing and Communications Conference. 2004: 445-452.

[5] Laoutaris N, Che H, and Stavrakakis I. The LCD interconnection of LRU caches and its analysis [J]. Performance Evaluation, 2006, 63 (7): 609-634.

[6] Psaras I, Chai W K, and Pavlou G. Probabilistic in-network caching for information-centric networks [C]// Proc of the 2nd Edition of the ICN Workshop on Information Centric Networking. 2012: 55-60.

[7] Chai W K, He D, Psaras I, et al. Cache"less for more"in information-centric networks (extended version) [J]. Computer Communications, 2013, 36 (7): 758-770.

[8] 蔡君，余顺争，刘外喜. 基于节点社团重要度的 ICN 缓存策略 [J]. 通信学报, 2015, 36 (6): 173-182.

[9] Ren J, Qi W, Westphal C, et al. MAGIC: a distributed max-gain in-network caching strategy in information-centric networks [C]// Proc of IEEE INFOCOM 2014. 2014: 470-475.

[10] Tsilopoulos C, Xylomenus G. Supporting diverse traffic types in information centric networks [C]// Proc of ACM SIGCOMM Workshop on Information Centric Networking. 2011: 13-18.

[11] 葛国栋，郭云飞，刘彩霞，等. CCN 中基于业务类型的多样化内容分发机制 [J]. 电子学报, 2016, 44 (5): 1124-1131.

[12] Zhang G, Wang X, Gao Q, et al. A hybrid ICN cache coordination scheme based on role division between cache nodes [C] Proc of IEEE Global Communications Conference. 2015: 1-6.

[13] 张果，胡宇翔，黄万伟，等. 基于流行内容感知和跟踪的协同缓存策略 [J]. 通信学报, 2017, 38 (2): 132-142.

[14] Montgomery D, Johnson D, Gardiner J. Forecasting and time series analysis [M]. New York: McGraw-Hill, 1990: 618-618.

[15] Afanasyev A, Moisseenko I, Zhang L. ndnSIM: NDN simulator for NS-3 [R]. [S. l.] : University of California Technical Report, 2012.

[16] Pittel B, Spencer J, Wormald N. Sudden emergence of a giant k-core in a random graph [J]. Journal of Combinatorial Theory, 1996, 67 (1): 111-151.

[17] Zhang G, Zhang J H, Wang B Q, et al. On-line popularity monitoring method based on bloom filters and hash tables for differentiated traffic [J]. China Communications, 2016, 13 (z1): 72-86.